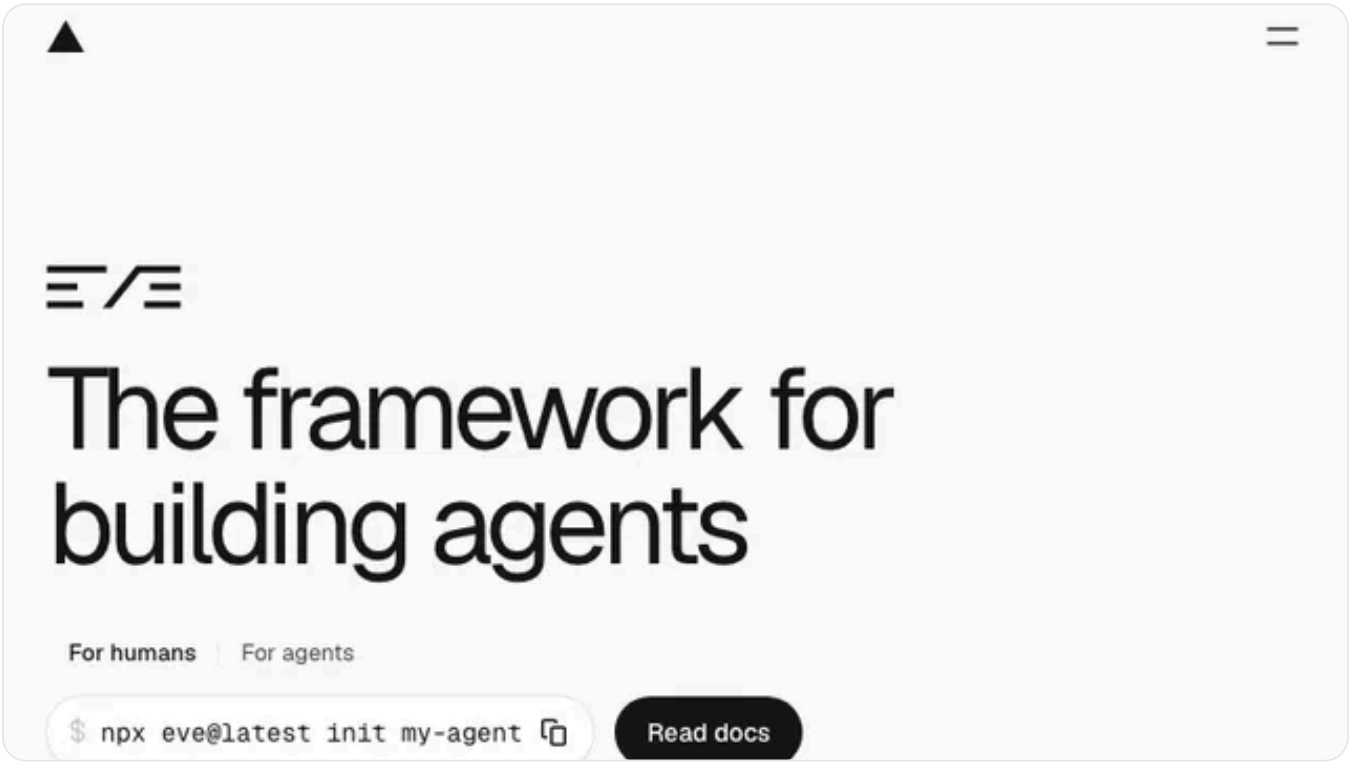
Vercel Eve

A framework for building durable AI agents with markdown instructions, TypeScript tools, skills, workflows, and channels.



About

Vercel Eve is an AI agent framework for building and deploying production agents. It treats an agent like a directory: instructions live in Markdown, tools live in TypeScript, and the runtime is designed to be durable by default. The project positions itself like Next.js for web apps, but for agents, with support for skills, workflows, sandboxes, and channels.

What is Vercel Eve

Vercel Eve is an AI agent framework from Vercel that helps you build production-ready agents with a directory-based structure. The core idea is simple: an agent can be

configuration, then compiled and deployed through the framework. Vercel describes it as “like Next.js for web apps, but for agents,” which makes the product feel familiar to anyone who has used Vercel’s app workflow before.

The app is centered on durable AI agent execution rather than one-off prompts. That means Vercel Eve is built for agents that need instructions, skills, tools, workflows, and channels to work together consistently over time. If you are looking for an AI agent framework that maps neatly onto a codebase, Eve is aimed at that use case.

Key features

Vercel Eve focuses on a small set of building blocks that are meant to scale from a simple prototype to a more complete agent system.

- **Markdown instructions**: the site shows an `instructions.md` file as the starting point for an agent.
- **TypeScript tools**: reusable tools are written in TypeScript, which keeps the tool layer programmable and testable.
- **Skills in Markdown**: skills are described as Markdown playbooks and are loaded when relevant.
- **Durable by default**: the framework highlights durability as a core behavior, which is important for long-running AI agents.
- **Workflows and channels**: the platform says it wires up durable workflows and connects channels as part of the compile and deploy process.
- **Model configuration**: the agent can use a default model, and an `agent.ts` file lets you choose a model or configure the runtime.
- **AI Gateway integration**: the source specifically notes that the model layer leverages AI Gateway.

These features make Vercel Eve an AI agent framework rather than a no-code chatbot builder. It is designed for developers who want to shape behavior with code and content, not just prompts.

How to use it

Run the starter command shown on the site, such as `npx eve@latest init my-agent`.

Add an `instructions.md` file to define the agent's role and behavior in Markdown.

Create `agent.ts` if you need to select a model or adjust runtime settings.

Add TypeScript tools in a `tools/` folder, such as a `get_weather.ts` example.

Add Markdown skills in a `skills/` folder when you want reusable guidance that loads only when needed.

Deploy the compiled agent after the directory is wired up by the framework.

A practical Eve setup can stay small at first. The site's example shows that an `instructions.md` file alone can represent a complete agent, while `agent.ts`, tools, and skills extend the system when the use case becomes more complex.

Who it is for

Vercel Eve is for developers who want to build an AI agent framework that feels structured, code-first, and deployable.

AI app developers who already work with TypeScript and want a framework that matches their stack.

Teams building production agents that need durable workflows instead of experimental prompts.

Vercel users who want a product aligned with the wider Vercel ecosystem.

Builders who prefer file-based organization over a visual low-code interface.

Teams that need reusable skills and tools for agents that must do more than answer questions.

It is also a fit for people who want a clearer separation between instructions, tools, and runtime configuration in an AI agent framework. That structure can make larger agents easier to maintain.

What to know before you use it

Vercel Eve is powerful, but there are a few limits and caveats to keep in mind before you rely on it as your AI agent framework.

The framework is not presented as a general chatbot app; it is aimed at developers building agents with code and Markdown.

Since tools are written in TypeScript, this is not a no-code experience.

The site mentions durability, workflows, channels, and AI Gateway, but it does not spell out every runtime guarantee or operational boundary in the source we have.

For anything that depends on exact model behavior, delivery guarantees, or account terms, use the official Vercel documentation and pricing pages instead of relying on a summary.

If you need a structured AI agent framework that uses Markdown for instructions, TypeScript for tools, and a directory-first workflow, Vercel Eve is built for that. If you need a fully documented commercial plan or complete operational spec, check Vercel's official docs before committing to production.

FAQ

What is Vercel Eve?

What is Vercel Eve used for?

How do I start an agent project with Vercel Eve?

Is Vercel Eve a no-code tool?

What programming model does Vercel Eve use?

Does Vercel Eve have pricing listed?

Is Vercel Eve free to use?

What do I need to start an agent in Eve?

What makes Vercel Eve different from other agent frameworks?

Who should use Vercel Eve?

Sponsored

Belay – Awa... Featured
Developer Tools

Stays awake while your agents work.

Free # Developer Tools

View ↗

Partly Featured
Finance

Partly is a mobile app that lets friends quickly scan receipts, split any bill fairl...

Free # Finance

View ↗

Visit website ↗

vercel-eve

Pricing	Free
Category	AI
Platform	Web, API
Launched	Aug 2026
Updated	4 days ago
Verified	2 months ago

TAGS

agent framework # ai-agents # vercel # automation # typescript
developer-tool # markdown # deployment # workflows

Dir Hub

A curated directory of small, well-made apps. Browse by category or search to find the right one in fewer clicks.

DISCOVER

Home

Pricing

Submit your app

COMPANY

About

Contact

LEGAL

Privacy

Terms

© 2026 Dir Hub

Curated indie apps

Vercel Eve